Docket No.   AUS920010995US1

CLAIMS:

What is claimed is:

1.   A method for creating a software state machine, comprising:

5   providing a state machine object; and

providing an initializer object, wherein the initializer object defines states, actions, and conditions for a software state machine,

wherein the state machine object is configured to

10   use the initializer object to create an array of state transition objects and execute the software state machine using the array of state transition objects.

2.   The method of claim 1, wherein the state machine object includes an object constructor method.

15   3.   The method of claim 2, wherein the object constructor method is configured to create an instance of the initializer object.

4.   The method of claim 1, wherein the state machine object is configured to create a table object.

20   5.   The method of claim 4, wherein the initializer object includes a table element array creation method.

6.   The method of claim 5, wherein the state machine object is configured to call the table element array creation method and create the table object using the

25   results of the table element array creation method.
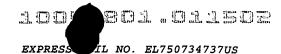
Docket No.   AUS920010995US1

7.   The method of claim 1, wherein the initializer object includes a table variable array creation method.

8.   The method of claim 7, wherein the state machine object is configured to call the table variable array creation method and create an array of state variables using the results of the table variable array creation method.

9.   The method of claim 1, wherein at least one of the state machine object and the initializer object implements an interface.

10.   The method of claim 1, wherein the state machine object includes a state method that is configured to return a current state of the software state machine.

11.   A method for creating software state machines, comprising:
        providing a state machine object;
        creating a first instance of the state machine object with a first state machine initializer, wherein the first instance of the state machine object executes a first software state machine; and
        creating a second instance of the state machine object with a second state machine initializer, wherein the second instance of the state machine object executes a second software state machine.

12.   An apparatus for creating a software state machine, comprising:
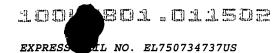
Docket No.   AUS920010995US1

a processor; and

a memory having stored therein a state machine object and an initializer object,

wherein the initializer object defines states,
5   actions, and conditions for a state machine,

wherein the state machine object is configured to use the initializer object to create an array of state transition objects and execute the software state machine using the array of state transition objects.

10   13.   The apparatus of claim 12, wherein the state machine object includes an object constructor method.

14.   The apparatus of claim 13, wherein the object constructor method is configured to create an instance of the initializer object.

15   15.   The apparatus of claim 12, wherein the state machine object is configured to create a table object.

16.   The apparatus of claim 15, wherein the initializer object includes a table element array creation method.

17.   The apparatus of claim 16, wherein the state machine
20   object is configured to call the table element array creation method and create the table object using the results of the table element array creation method.

18.   The apparatus of claim 12, wherein the initializer object includes a table variable array creation method.
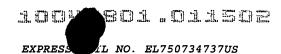
Docket No.   AUS920010995US1

19.   The apparatus of claim 18, wherein the state machine object is configured to call the table variable array creation method and create an array of state variables using the results of the table variable array creation

5   method.

20.   The apparatus of claim 12, wherein at least one of the state machine object and the initializer object implements an interface.

21.   The apparatus of claim 12, wherein the state machine

10   object includes a state method that is configured to return a current state of the software state machine.

22.   An apparatus, comprising:

a state machine initializer object;

a state machine object; and

15   a virtual machine, wherein the virtual machine creates an instance of the state machine object including an object constructor method,

wherein the object constructor method creates an instance of the initializer object and uses the instance

20   of the initializer object to create a table object and an array of state variables,

wherein the table object includes a state array creation method and the object constructor method calls the state array creation method to create an array of

25   state transition object, and

wherein the instance of the state machine object uses the array of state transition objects to execute a software state machine.

Docket No.   AUS920010995US1

23.   A computer program product, in a computer readable
medium, for creating a software state machine,
comprising:

   instructions for providing a state machine object;
5 and

   instructions for providing an initializer object,
wherein the initializer object defines states, actions,
and conditions for a software state machine,

   wherein the state machine object is configured to
10 use the initializer object to create an array of state
transition objects, receive the array of state transition
objects, and execute the software state machine.

24.   A computer program product, in a computer readable
medium, for creating software state machines, comprising:
15   instructions for creating a first instance of a
state machine object with a first state machine
initializer, wherein the first instance of the state
machine object executes a first software state machine;
and
20   instructions for creating a second instance of the
state machine object with a second state machine
initializer, wherein the second instance of the state
machine object executes a second software state machine.